

REMARKS

Applicants amended claims 1, 12, and 23, as discussed below.

Applicants added claims 34-39.

Claims 34, 36, and 38 depend from claims 1, 12, and 23, respectively, and further require that the attribute information describes whether the symbol definition or symbol reference is to data or executable code. The additional requirements of these claims are disclosed on at least page 3 and 11-12 of the Application.

Claims 35, 37, and 39 depend from claims 34, 36, and 38, respectively, and further require that the attribute information indicates data types and structures if the symbol definition or symbol reference is for data and indicates a number and types of arguments passed or received if the symbol definition or symbol reference is for executable code. The additional requirements of these claims are disclosed on at least page 3 and 11-12 of the Application.

1. Claims 1, 3, 5-7, 12, 14, 16-18, 23, 25, and 27-29 are Patentable Over the Cited Art

The Examiner rejected claims 1, 3, 5-7, 12, 14, 16-18, 23, 25, and 27-29 as anticipated (35 U.S.C. §102) by Lee (U.S. Patent No. 5,553,286). Applicants traverse this rejection for the following reasons.

Claims 1, 12, and 23 concern producing an executable file for execution by a computer, comprising: receiving a plurality of programming language statements comprising a source program into the computer, wherein the language statements include declarations defining attribute information for symbol references and symbol definitions; translating the source program into an object module, wherein the object module is capable of including: a symbol reference; a symbol definition; attribute information for the symbol reference derived from the language statements declaring the attribute information; and attribute information for the symbol definition derived from the language statements declaring the attribute information; binding object modules into a program object, wherein the attribute information is available when binding object modules into the program object; resolving in the program object an external symbol reference in the object module with an external symbol definition in another object module.

Applicants amended claims 1, 12, and 23 to further require that the language statements include declarations defining attribute information for symbol references or definitions, and that the attribute information for the symbol reference or symbol definition are derived from the language statements declaring the attribute information.

The Examiner cited col. 6, line 51 to col. 7, line 1 and col. 8, lines 37-37 of Lee as disclosing the claim requirement that the object module include attribute information for the symbol reference derived from the language statements; and attribute information for the symbol definition derived from the language statements. (Final Office Action, pgs. 2, 9) Applicants traverse for the following reasons.

The Examiner found that the index numbers, size and offset fields discussed in cited cols. 6-7 disclose the claimed attribute information for the symbol reference and definition. The cited index, size and offset information of Lee concern the physical limitations of a load module. Lee discusses that source programs are compiled into object modules, which are linked together to form a load module, which is executable. (Lee, col. 6, lines 9-19). Thus, the cited size and offset fields concern physical limitations of the load module that are eliminated in the program object to allow accommodation of more sections and modules in the program object of Lee. Such cited information on the physical limitations of a program object with respect to the sections and modules included in the program object having nothing to do with and do not disclose attribute information on symbol references and definitions included in the object module that is derived from the language statements that define such attribute information.

Further, this cited information of Lee on the physical layout of a program object does not disclose the claimed attribute information for a symbol reference or definition that provides information defined in the source program language statements.

In other words, Applicants submit that the claimed symbol reference or definition attribute information included in the object module is different from the cited offset and size fields that define physical limitations of how object modules may be included in the program object of Lee.

The Examiner also found that the pointers discussed in cols. 6-7 of Lee discloses the claimed symbol attribute information. (Final Office Action, pg. 9). Applicants traverse because a pointer is

not attribute information on a symbol reference or definition defined in the language statements that declare such attribute information as claimed. Instead, a pointer is just a way to locate an item. Thus, the discussion in the cited col. 6 of replacing external names from a dictionary with pointers does not disclose the claimed symbol attribute information that is derived from the language statements declaring and defining the attribute information.

Further, the index numbers the Examiner cites in col. 6, lines 55-65 also do not disclose the claimed attribute information. (Final Office Action, pgs. 9-10) These cited index numbers are an artifact of the program object and concern the size of numbers used to locate external names and sections. Such index numbers concerning the program object do not disclose the claimed symbol attribute information that is derived from the language statements declaring the attribute information. Further, Applicants submit that the cited index number do not comprise the claimed attribute information that provides information defined in the source program language statements.

Moreover, the above discussed size, offset, pointers, and index numbers providing information on the layout of a program object cannot comprise the claimed attribute information because the claims require that the defined attribute information for the symbol reference or definition is included in the object module. Information on the layout of a program object in which object modules are included does not disclose or comprise attribute information on a symbol included in an object module.

The cited col. 8 discusses how the binder overlays modules and other items to form the single program object. This section discusses information the binder uses to construct the program object, including what is referred to as class attributes. Applicants submit that this information the binder uses to construct a program object is different from the claimed symbol reference or attribute information included in the object module that is derived from the language statements defining.

Thus, the cited Lee nowhere discloses the claim requirement that the object module include attribute information on a symbol reference or definition that is derived from the language statements that declare and define such attribute information for the symbols.

The Examiner cited col. 7, lines 4-30 and col. 8, lines 37-46 as disclosing the claim requirement of binding object modules into a program object, wherein the attribute information is

available when binding object modules into the program object. (Final Office Action, pgs. 2, 10)
Applicants traverse.

The cited col. 8 discusses how the binder overlays items to form the single program object. The cited col. 7 discusses information used by the binder when constructing the program object, such as class attributes. Applicants submit that the cited information and techniques the binder uses to construct a program object nowhere discloses the claim requirement that attribute information on a symbol reference or definitions, derived from the language statements that declare and define such attribute information and included in the object module, is available to the binder when binding object modules into the program object. Instead, the cited attributes the binder uses to control the loading and binding of items in cols. 7 and 8 concern different types of information.

Accordingly, claims 1, 12, and 23 are patentable over the cited art because the cited Lee does not disclose all the claim requirements.

Claims 3, 5-7, 14, 16-18, 25, and 27-29 are patentable over the cited art because they depend from one of claims 1, 12, and 23, either directly or indirectly. The following dependent claims provide further grounds of patentability over the cited art

Claims 3, 14, and 25 depend from claims 1, 12, and 23 and further require that the object module is further capable of including fixed attribute information derived from language statements declaring attribute information for the symbol reference and symbol definition.

The Examiner cited col. 7, lines 4-30 and 47-49 and col. 9, lines 38-45 of Lee as disclosing the additional requirements of these claims. (Final Office action, pgs. 3 and 10-11) Applicants traverse.

The cited col. 7, lines 4-30 discuss class attribute information that controls the loading and binding of items in the class, including physical organization, how the items are to be structured during binding, whether the class can contain address constants, whether the class should be loaded into storage, etc. The cited col. 7, lines 47-49 mentions that the binder logic is reduced by confining loading and binding variations to a finite set of class attributes.

As discussed, the cited col. 7 concerns information the binder uses when overlaying object modules into the program object of Lee. Nowhere does this cited col. 7 anywhere disclose the claim

requirement of the object module including fixed attribute information derived from language statements declaring attribute information for the symbol reference or definition that is available during binding.

The cited col. 9 mentions that the loader opens a file and that the object's structural data is read into memory. The loader allocates storage for each loadable segment in the object. The length, location and other characteristics of each block of storage will be determined by the class attributes. This cited section concerns how to load a program object and nowhere discloses the claim requirements that the attributes available to the binder include fixed attribute information derived from language statements declaring attribute information for a symbol reference or definition

Accordingly, claims 3, 14, and 25 provide additional grounds of patentability over the cited art because the cited Lee does not disclose the additional dependent claim requirements.

Claims 5, 16, and 27 depend from claims 1, 12, and 23 and require that the object module is further capable of including an address constant for a symbol referenced in the module and attribute information derived from language statements declaring attribute information for the address constant.

The Examiner cited col. 8, lines 48-52 and col. 3, lines 16-18 as disclosing the claim requirement of attribute information derived from language statements declaring attribute information for the address constant, where the attribute information is for symbol definitions and references. (Final Office Action, pgs. 3, 11-12). Applicants traverse.

The cited col. 3 discusses the length and placement of a text record. The cited col. 8 mentions that the binder stores the target address of a relocatable address constant and that class offsets are stored in the address constant. Although the cited Lee discusses address constants, nowhere does the cited col. 8 anywhere disclose the claim requirement that attribute information for a symbol reference and definition is derived from language statements declaring attribute information for the address constant.

The Examiner further cited the relocation dictionary discussed in col. 3, lines 14-15 that includes the location and type of each address constant in a text record. (Final Office Action, pgs. 11-12). This information maintained in a relocation directory on an address constant is different from and does not disclose that the object module include attribute information for a symbol reference or

definition derived from language statements declaring and defining such attribute information for the address constant.

Further, the cited relocation directory is a record in the load module. (Lee, col. 3, lines 2-20). Claims 5, 16, and 27 concern attribute information in the object module derived from the language statements, which is different from the records of the load module in which object modules are included.

Accordingly, claims 5, 16, and 27 provide additional grounds of patentability over the cited art because the cited Lee does not disclose the additional dependent claim requirements.

Claims 6 and 7; 17 and 18; and 28 and 29 depend from claims 5, 16, and 27, respectively, and provide additional requirements concerning the requirement that address constants provide attribute information for the symbol references. Again, although the cited Lee discusses address constants, nowhere does Lee anywhere disclose that the object module include attribute information derived from the language statements for an address constant. For these reasons, claims 6, 7, 17, 18, 28, and 29 provide still further grounds of patentability over the cited art because they concern further requirements on the claimed address constant and symbol attribute information that is not disclosed in the cited Lee.

2. Claims 2, 4, 8, 9, 13, 15, 19, 20, 24, 26, 30, and 31 are Patentable Over the Cited Art

The Examiner rejected claims 2, 4, 8, 9, 13, 15, 19, 20, 24, 26, 30, and 31 as obvious (35 U.S.C. §103) over Lee in view of Fitzgerald (U.S. Patent No. 5,408,665). Applicants traverse for the following reasons.

Claims 2, 4, 8, 9, 13, 15, 19, 20, 24, 26, 30, and 31 are patentable over the cited combination because they depend either directly or indirectly from one of claims 1, 12, and 23, which are patentable over the cited art for the reasons discussed above.

Claims 2, 13, and 24 depend from claims 1, 12, and 23 and further require that the language statement is capable of indirectly declaring extended attribute information defined in another location in the object module.

The Examiner cited col. 10, line 47 to col. 11, line 35 of Fitzgerald as disclosing these claim requirements. (Final Office action, pgs. 4-5, 12-13) Applicants traverse.

The cited cols. 10-11 of Fitzgerald discuss an extended dictionary that includes information on object modules in a library to locate the object modules in the library. Applicants submit that this cited information in Fitzgerald to locate object modules using an extended dictionary, such as the Module ID, does not teach or suggest the claim requirement that the language statement from which the symbol attribute information is derived is indirectly declared in another location in the object module. In other words, information in the extended dictionary used to locate object modules in a library is different and does not disclose the claimed extended attribute information declared in the language statements and defined in another location of the object module.

Accordingly, claims 2, 13, and 24 provide additional grounds of patentability over the cited art.

Claims 8, 19, and 30 depend from claims 1, 12, and 23 and further require the object module further includes an External Symbol Directory (ESD) including a record capable of indicating a symbol in the program, a location of the symbol in the program, and a pointer to attribute information in the program for the symbol.

The Examiner cites the External Dictionary of Fitzgerald as teaching the claim requirement that the ESD include a pointer to attribute information in the program for the symbol. (Final Office Action, pgs. 5-6) Applicants traverse.

As discussed, the cited External Dictionary of Fitzgerald includes information to locate modules. The Examiner has not cited any part of Fitzgerald that discloses the claim requirement of a pointer to attribute information for a symbol definition or reference, where such attribute information is derived from language statements that include declarations of attribute information that provides information defined in the source program language statements. Instead, the cited External Dictionary concerns information on how to locate modules. Fitzgerald discusses how this External Dictionary is used by a linker to locate object modules to link together. (Fitzgerald, col. 8, line 62 to col. 10, line 9). Fitzgerald discusses how the Extended Dictionary allows object modules to be located during the passes when linking. (Fitzgerald, col. 9, line 49 to col. 10, line 9). The cited

figures 4b, 5b, 6b, c, d concern steps to process the extended dictionary during linking passes to process the object modules. Nowhere does the cited Fitzgerald anywhere teach or suggest the claimed pointer to attribute information for a symbol definition or reference as claimed.

Accordingly, claims 8, 19, and 30 provide additional grounds of patentability over the cited art.

Claims 9, 20, and 31 depend from claims 1, 12, and 23 and further require that the object module further includes a Relocation List Directory (RLD) including a record capable of indicating the location of the referenced symbol, a location of an address constant for the referenced symbol in the program, and a pointer to attribute information for the address constant in the program.

The Examiner again cited the External Dictionary of Fitzgerald discussed above as teaching the claim requirement a pointer to attribute information for the address constant in the program. (Final Office Action, pgs. 6-7) Applicants traverse.

The Examiner cited step 603 in FIGs. 6A. This step is part of the processing of the Extended Dictionary to locate an object module and to obtain an ID of the module needed by the current module. (Fitzgerald, col. 15, lines 35-40, as well as other steps mentioned in cols. 14-15)

Nowhere does the cited Fitzgerald anywhere teach or suggest the claim requirement of a pointer in the Relocation List Directory (RLD) to attribute information for the address constant in the program, where such attribute information is derived from language statements that include declarations of attribute information that provides information defined in the source program language statements.

Accordingly, claims 9, 20, and 31 provide additional grounds of patentability over the cited art.

3. New Claims 34-39 Are Patentable Over the Cited Art

New claims 34-39 are patentable over the cited art because they depend, either directly or indirectly, from one of claims 1, 12, and 23, which are patentable over the cited art for the reasons discussed above. Moreover, the additional requirements of these new claims in combination with the base and any intervening claims provide further grounds of patentability over the cited art.

4. Conclusion

For all the above reasons, Applicant submits that the pending claims 1-39 are patentable over the art of record. Applicants submit herewith the RCE for the RCE filed concurrently herewith and the fee for the added claims. Nonetheless, should any additional fees be required, please charge Deposit Account No. 09-0460.

The attorney of record invites the Examiner to contact him at (310) 553-7977 if the Examiner believes such contact would advance the prosecution of the case.

Dated: September 17, 2003

By: _____


David W. Victor

Registration No. 39,867

Please direct all correspondences to:

David Victor
Konrad Raynes Victor & Mann, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-553-7977
Fax: 310-556-7984